

FIG. 1

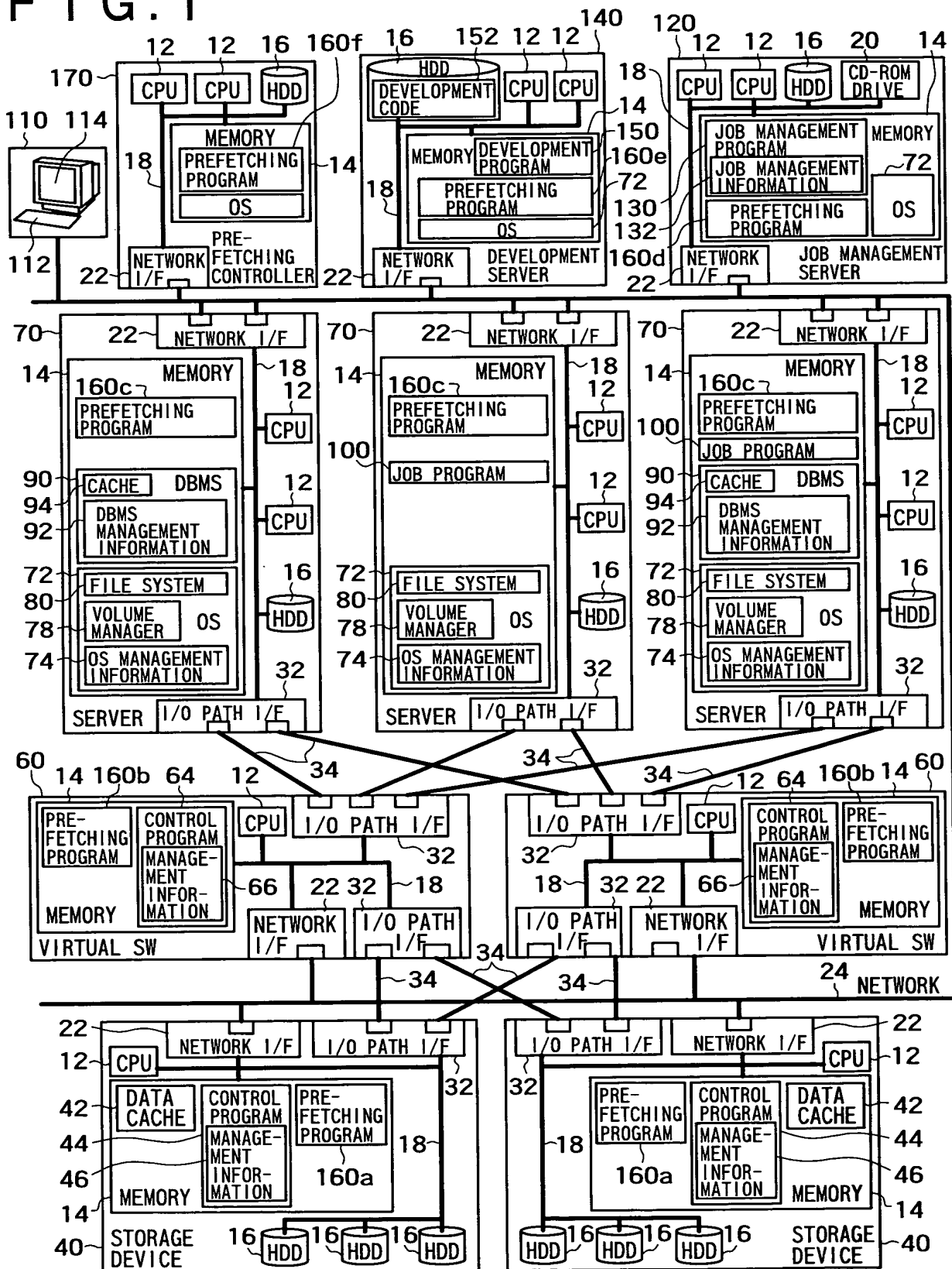


FIG. 2

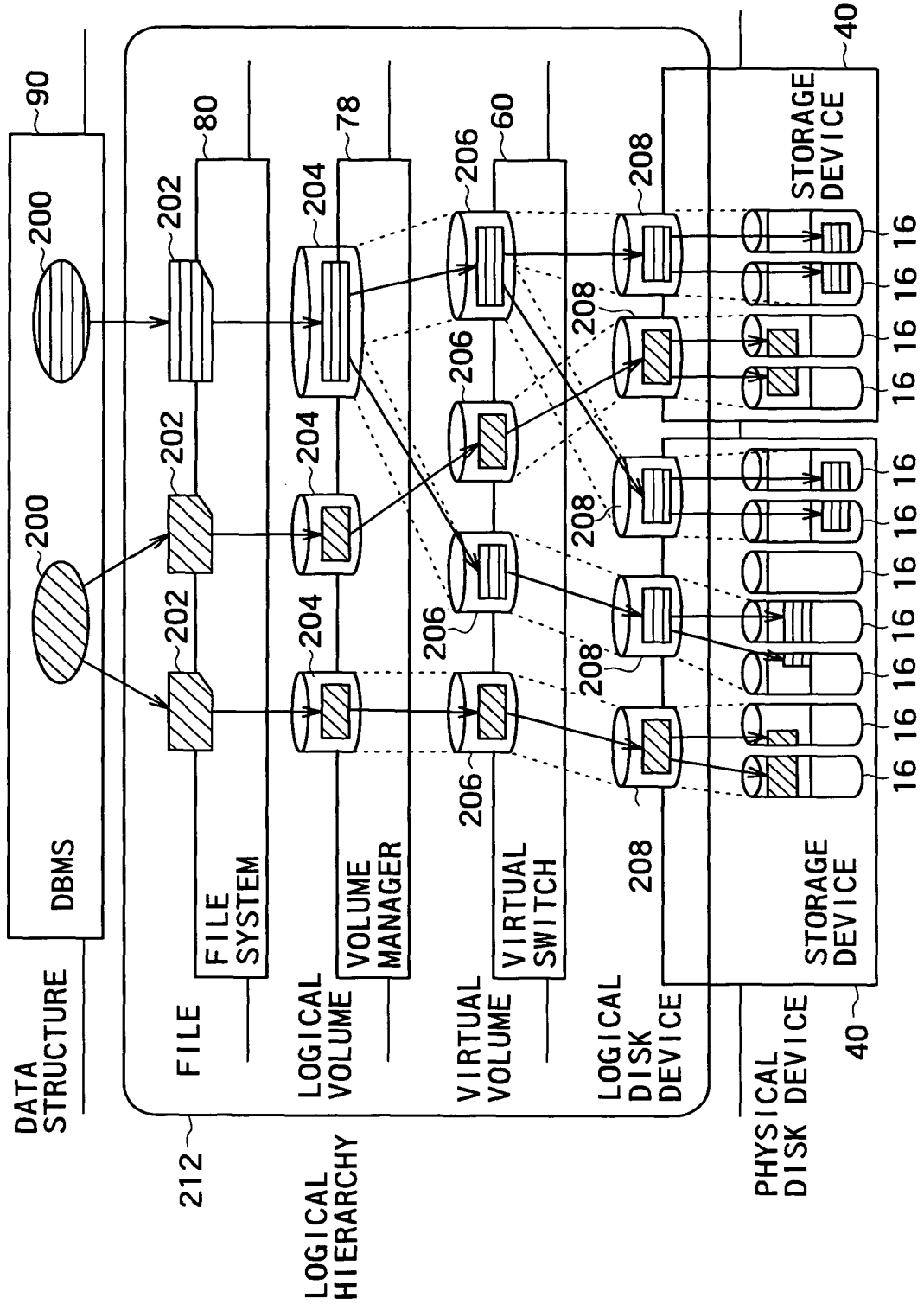


FIG. 3

312		314		310	
AREA INFORMATION OF UPPER VIRTUAL STRUCTURE		CORRESPONDING AREA INFORMATION OF LOWER MANAGEMENT STRUCTURE			
VIRTUAL STRUCTURE ID	INTER-STRUCTURAL AREA	VIRTUALIZING MACHANISM ID	MANAGEMENT STRUCTURE ID	INTER-STRUCTURAL AREA	
Upper0	0-10239	LowApp0	Lower0	0-10239	
	10240-20479	LowApp1	Lower1	0-10239	
Upper1	0-10239	LowApp0	Lower0	10240-20479	
	10240-20479	LowApp1	Lower1	10240-20479	
Upper2	0-10239	LowApp2	Lower0	0-10239	
	10240-20479	LowApp2	Lower1	0-10239	
⋮	⋮	⋮	⋮	⋮	
		AREA MAPPING INFORMATION			

FIG. 4

NAME OF DATA STRUCTURE	DATA STORAGE POSITION	
	NAME OF DATA FILE PATH	BLOCK NUMBER
T1	/dev/rdisk/Ivol0	0-499
	/dev/rdisk/Ivol0	2000-2499
T2	/dev/rdisk/Ivol0	500-799
T3	/dev/rdisk/Ivol1	500-799
⋮	⋮	⋮

DATA STORAGE AREA INFORMATION

FIG. 5

NAME OF DATA STRUCTURE	T1	T2	...	521
DATA PAGE SIZE	16KB	8KB	...	522
NUMBER OF DATA PAGES	61897	78783	...	524
CACHE QUANTITY	1024 PAGES	2048 PAGES	...	526

TABLE DATA QUANTITY INFORMATION

520

FIG. 6

NAME OF DATA STRUCTURE	Ind1-1	Ind2-1	...
NAME OF CORRESPONDING TABLE	T1	T2	...
TYPE OF INDEX	B-Tree	BitMap	...
DATA PAGE SIZE	8kB	8kB	...
NUMBER OF DATA PAGES	3782	6782	...
NUMBER OF LEAF NODE PAGES	2898	-	...
CACHE QUANTITY	1452 PAGES	50 PAGES	...
RETRIEVAL ATTRIBUTE	ID1 (ID1, ID2)	ID2	...
EXPECTATION TUPLE NUMBER	5.5	1	2058.2
INDEX INFORMATION			

FIG. 7

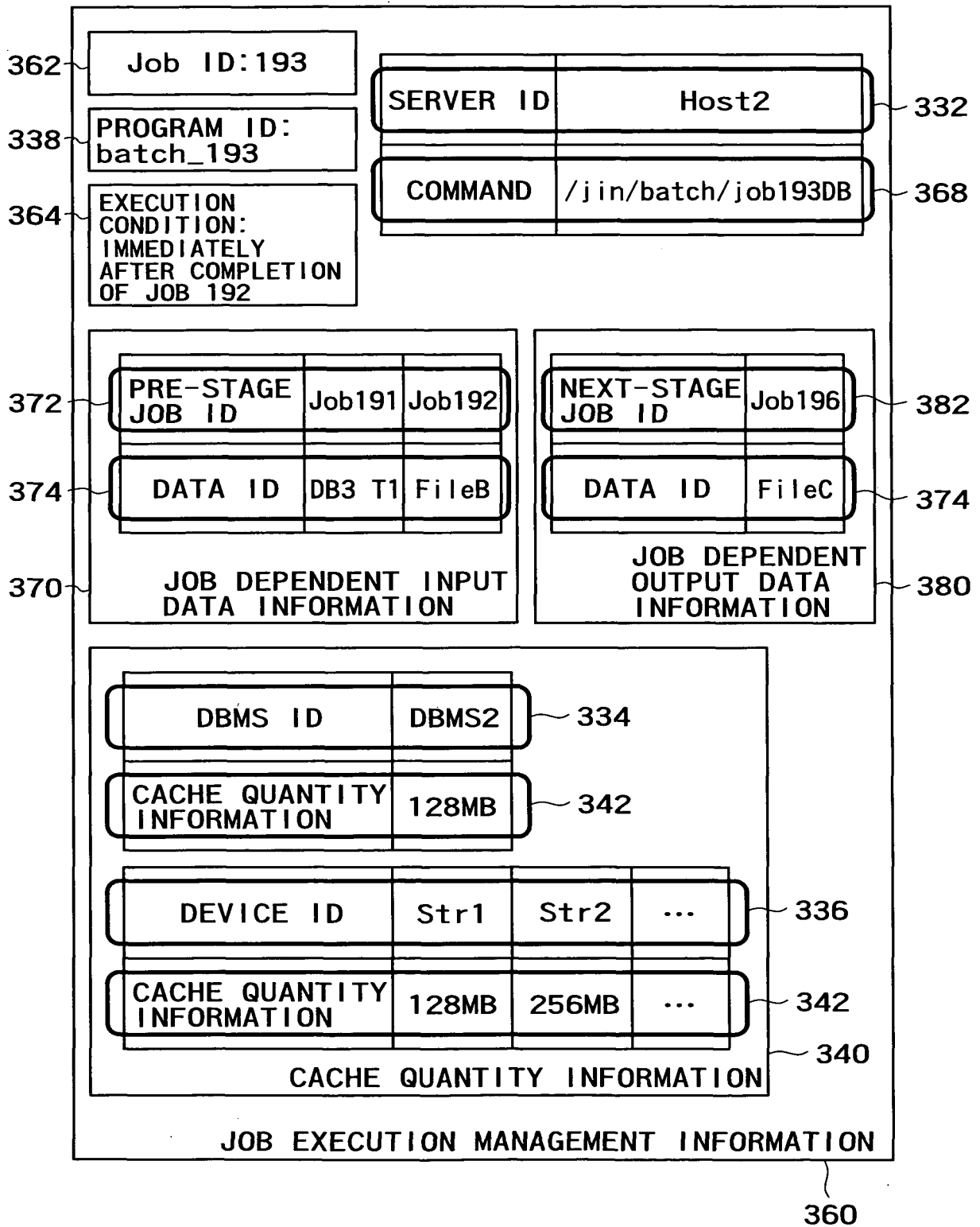


FIG. 8

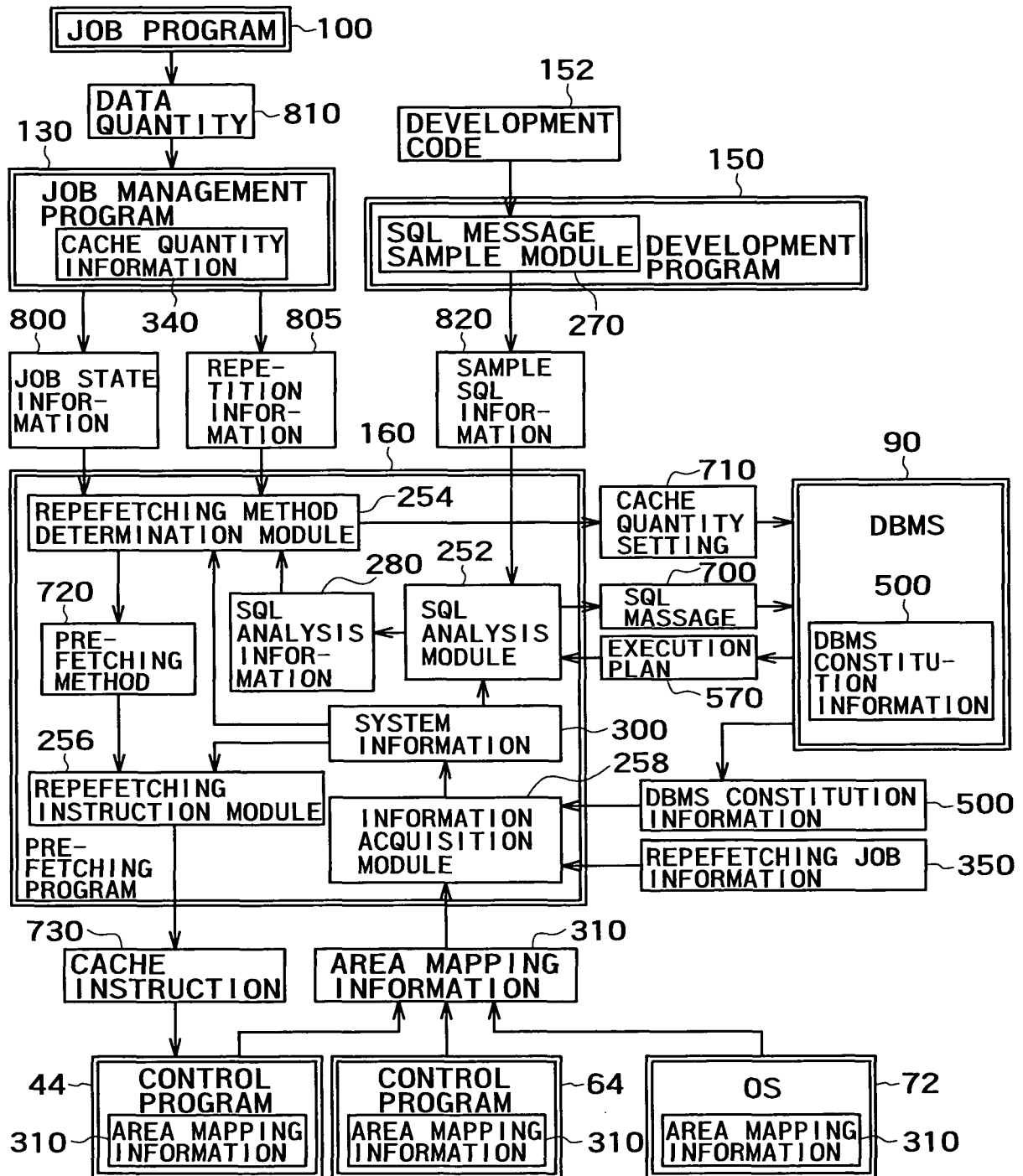


FIG. 9

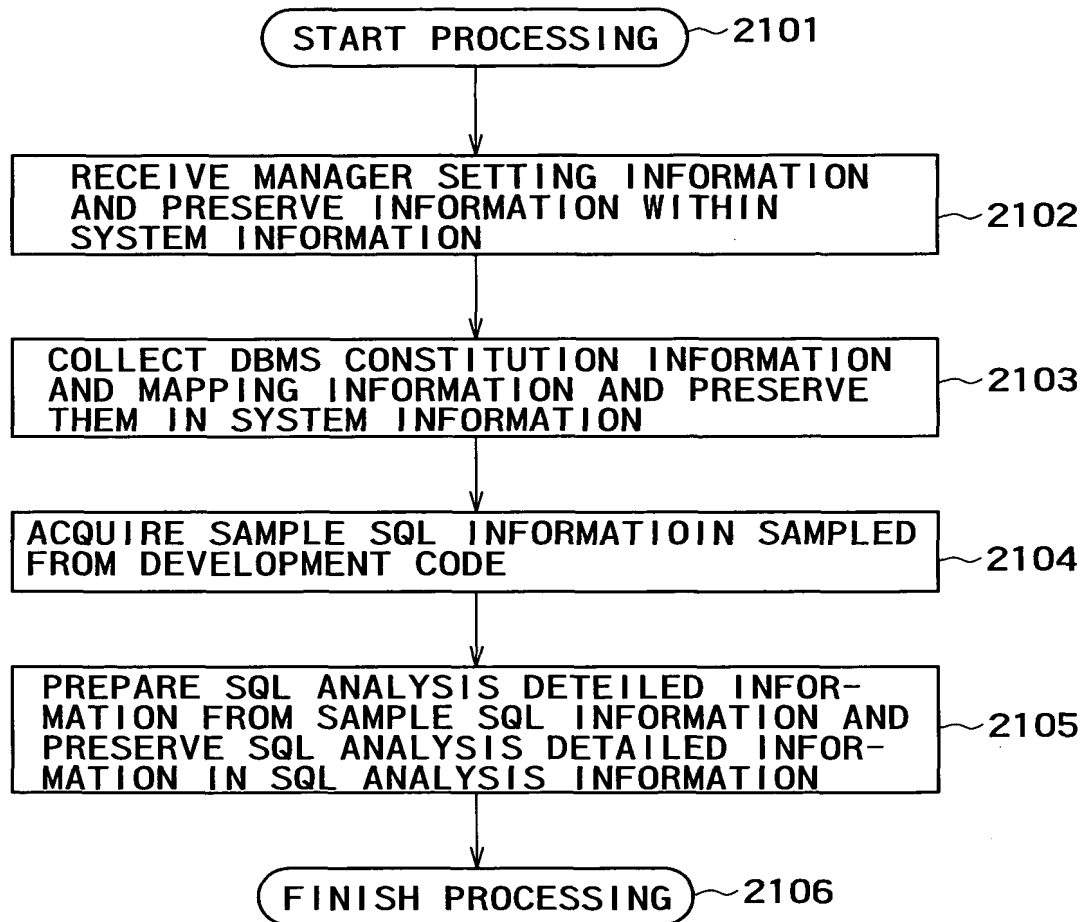
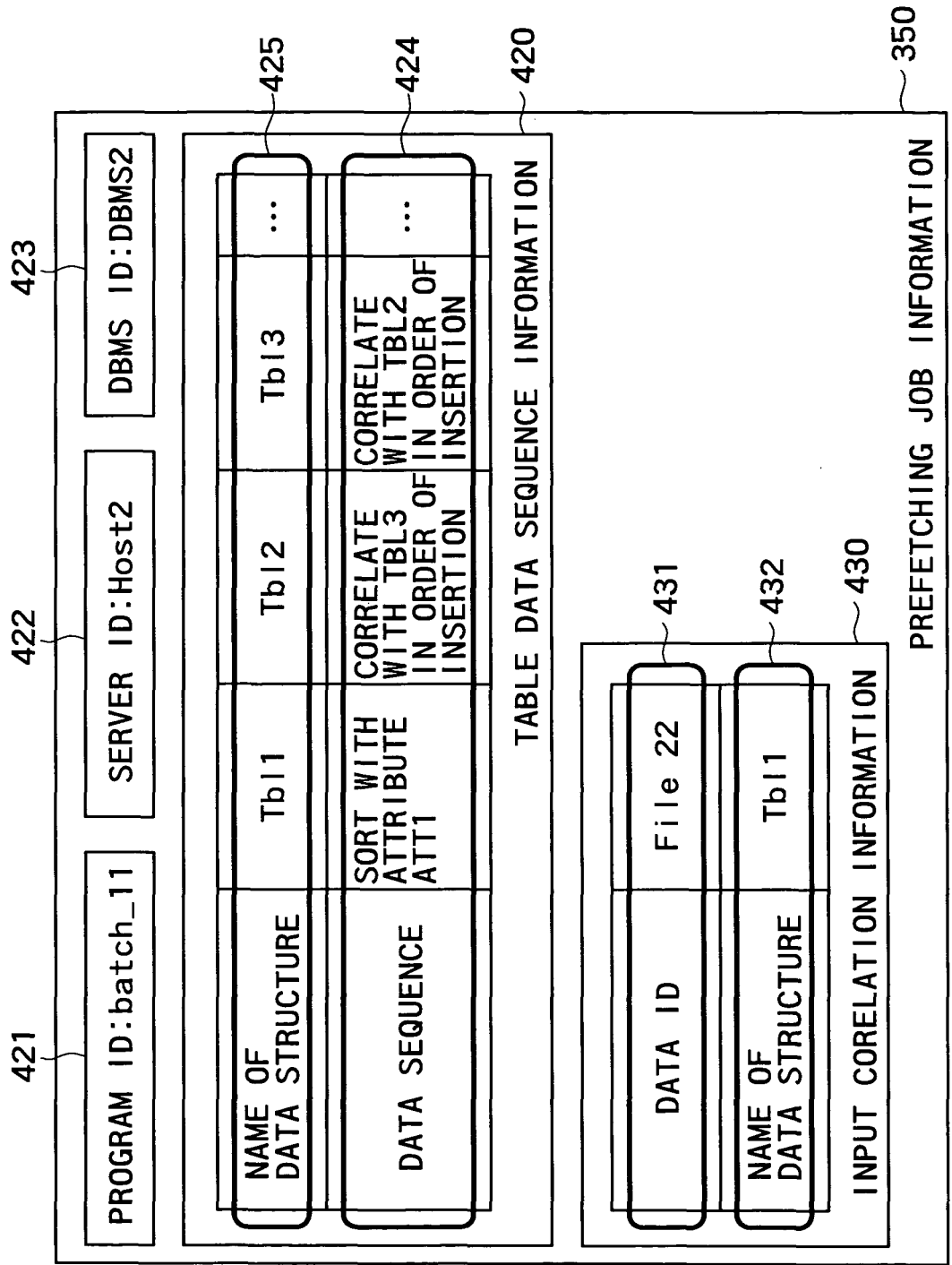


FIG. 10



10/29

FIG. 11

...

```
for(i=0;i<records;i++){  
...  
name=in_name[i]; addr=in_addr[i]; pay=i[i];  
EXEC SQL SELECT C_ID,C_ID INTO:c_id,:c_d_id  
FROM Cust  
WHERE C_Name=:name and C_Addr=addr  
FOR UPDATE;  
  
EXEC SQL UPDATE Cust  
SET C_Balance=C_Balance-:pay  
WHERE C_ID=:c_id;  
...  
COMMIT;  
}
```

5002

...

SAMPLING PROCESSING

...

5014

HINT REPEAT START LABEL(L_records_1)

5012

```
SELECT C_ID,C_ID  
FROM Cust  
WHERE C_Name=:name and C_Addr=addr  
FOR UPDATE;
```

```
UPDATE Cust  
SET C_Balance=C_Balance-:pay  
WHERE C_ID=:c_id;  
...  
COMMIT;
```

5010

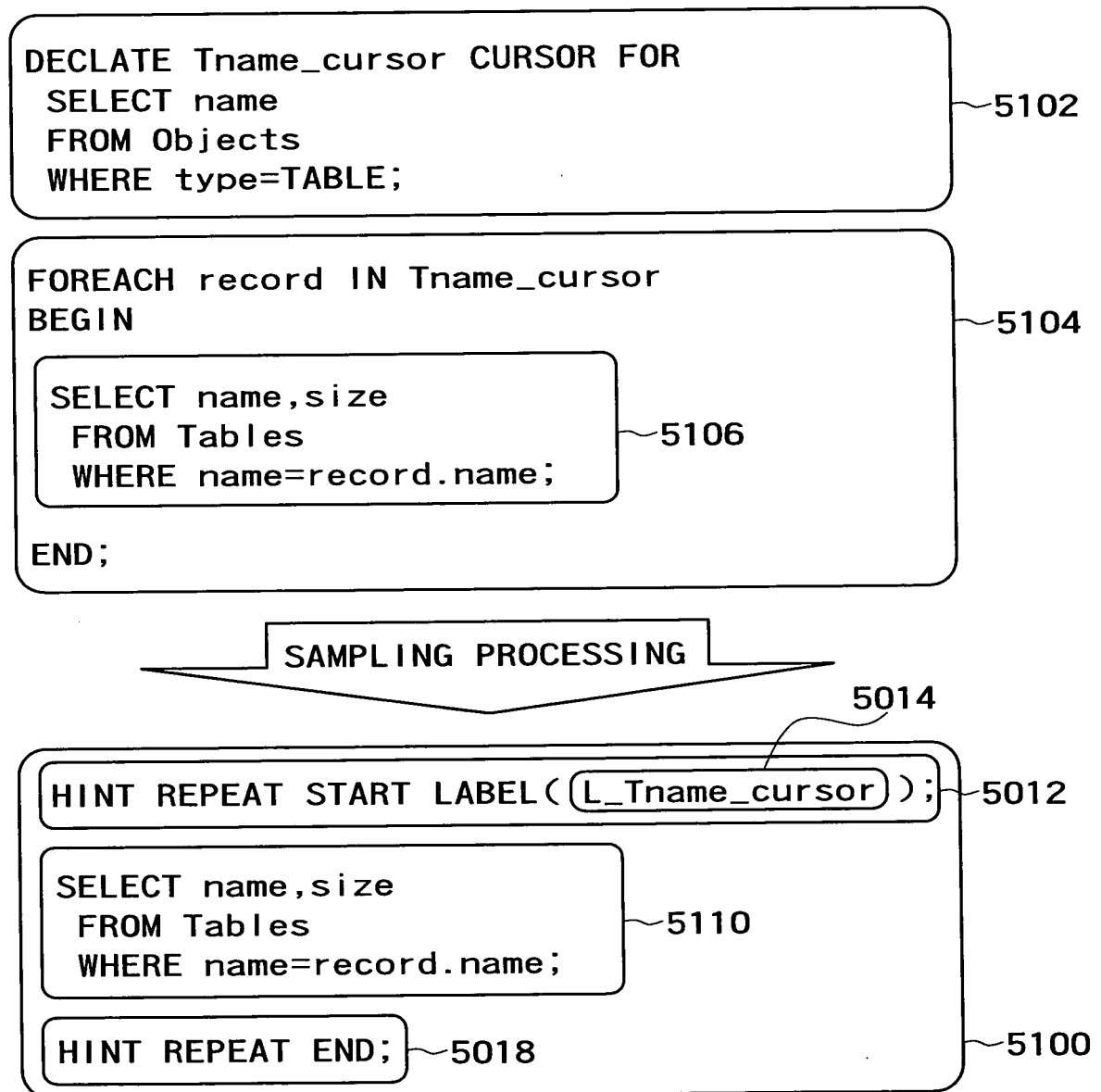
HINT REPEAT END;

5018

5000

...

FIG. 12



12/29

FIG. 13

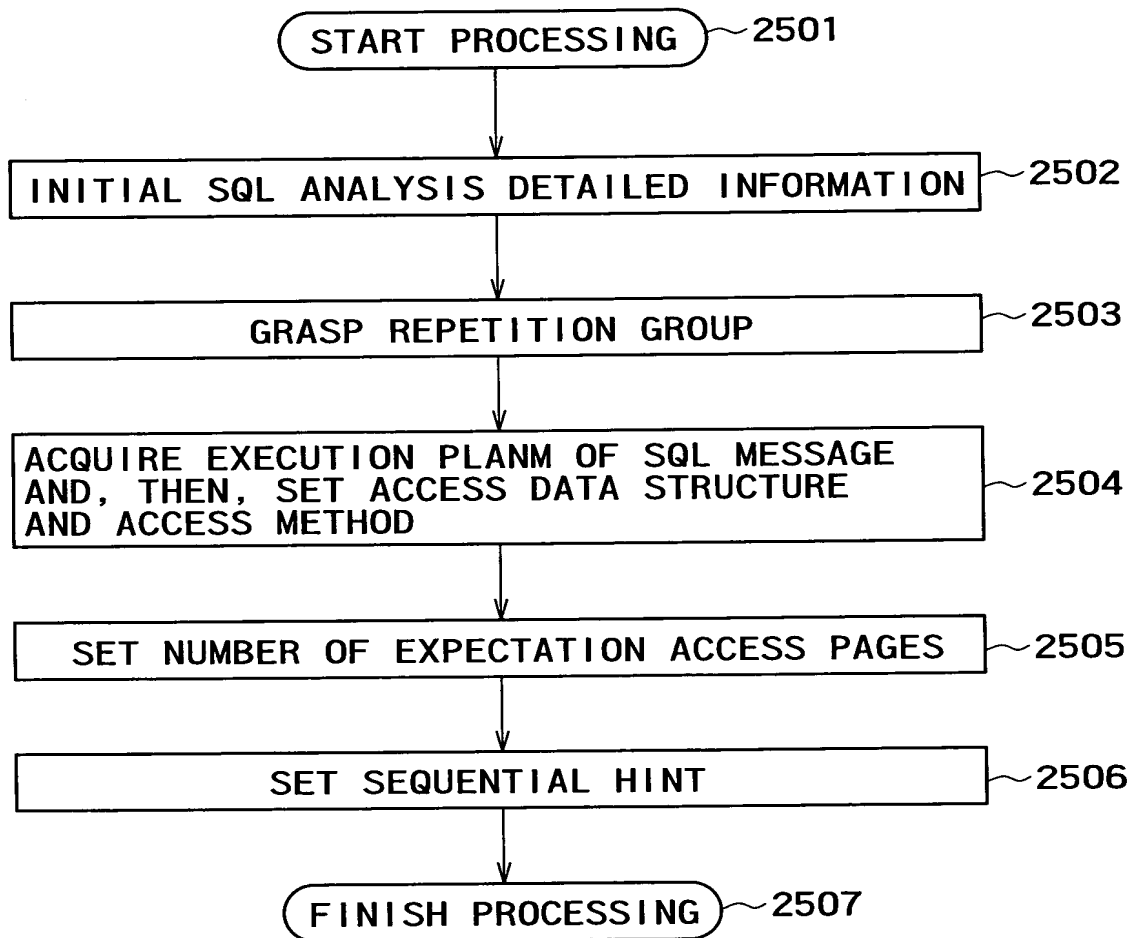


FIG. 14

PROGRAM ID:batch_11	DBMS ID:DBMS3	
REPETITION GROUP ID	L_loop_1	...
ORDER OF EXECUTION	1	...
DRIVE DATA ID	DB8 T1	...
NAME OF DATA STRUCTURE	T1	...
ACCESS METHOD	Table Access Full	Index Access
NUMBER OF EXECUTION ACCESS PAGE	-	4.5
SEQUENTIAL HINT	Y	-
		...

SQL ANALYSIS DETAILED INFORMATION

FIG. 15

572	574	576	578	582
NAME OF NODE	NAME OF PARENT NODE	CONTENT OF NODE PROCESSING	ACCESS DATA STRUCTURE	DETAIL OF NODE PROCESSING
N-1-1	root	Sum	-	Sum(T1, B)
N-2-1	N-1-1	Nested Loop Join	-	T1, Key1=T3, Key1
N-3-1	N-2-1	Hash Join	-	T3, Key3=T4, Key4
N-3-2	N-2-1	Table Access by Index	T1	-
N-4-1	N-3-1	Table Access Full	T3	-
N-4-2	N-3-1	Filter	-	T4, M<100
N-4-3	N-3-2	Index Access	Ind1-1	Key1=[RESULT OF N-3-1]
N-5-1	N-4-2	Table Access Full	T4	-
EXECUTION PLAN				
570				

FIG. 16

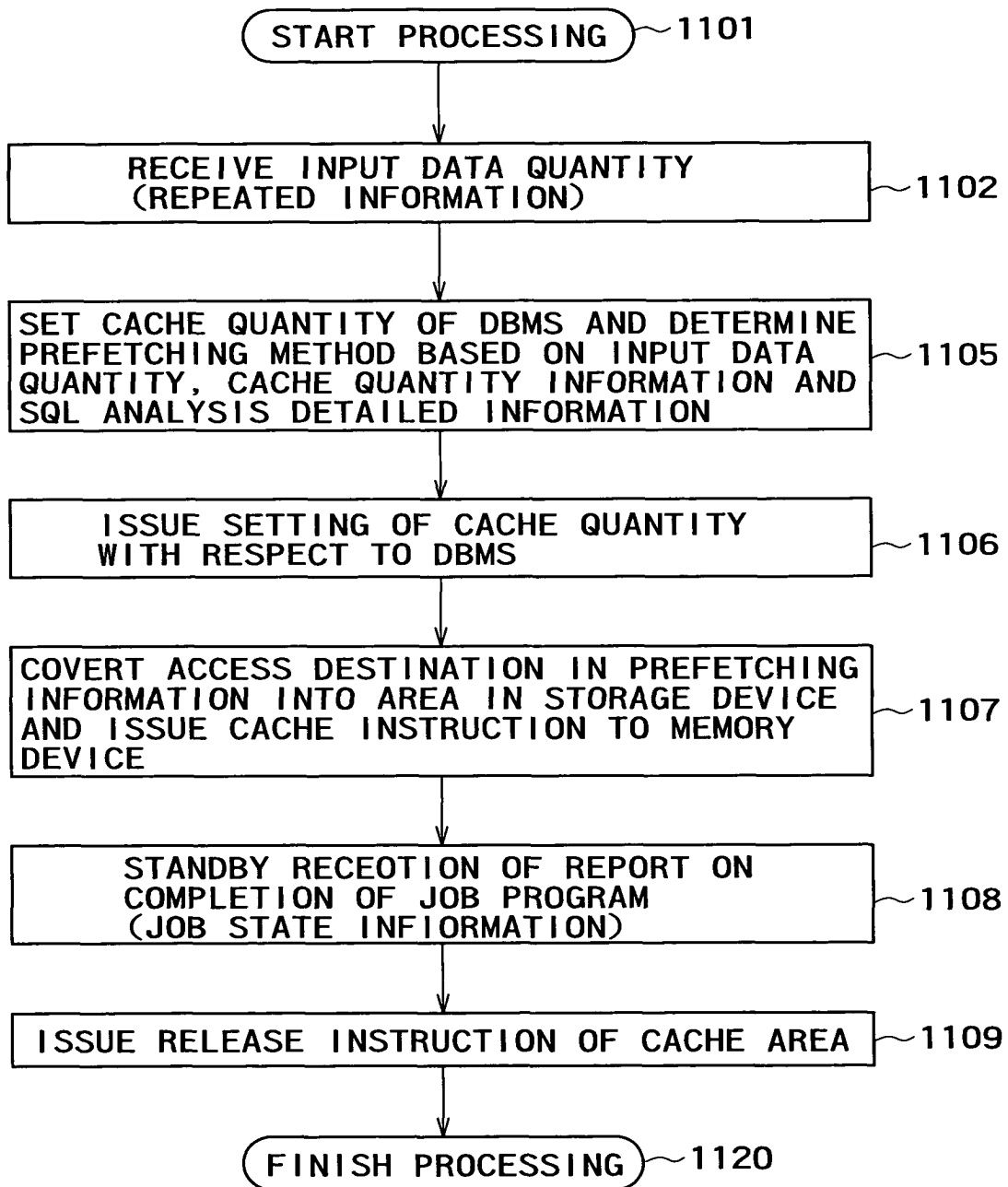


FIG. 17

NAME OF DATA STRUCTURE	T1	T2	...	711
CACHE QUANTITY	1024 PAGES	2048 PAGES	...	712

SETTING OF CACHE QUANTITY

710

FIG. 18

NAME OF DATA STRUCTURE	T1	Ind2-1	...	721
CACHE METHOD	SEQUEN-TIAL	INSTANT PREFETCHING	...	722
DRIVE ID	Str2	Str1	...	723
CACHE QUANTITY	4MB	40MB	...	724
ACCESS ORDER	1	2	...	725

PREFETCHING METHOD

720

FIG. 19

GROUPING		1	2	...
DATA AREA	VIRTUAL STRUCTURE IDENTIFIER	LU1	LU2	...
	AREA	10240-819199	81920-163839	...
CACHE METHOD		SEQUENTIAL	INSTANT PREFETCHING	...
CACHE QUANTITY		4MB	40MB	...
ACCESS ORDER		1	2	...

CACHE INSTRUCTION

730

732

734

735

736

737

FIG. 20

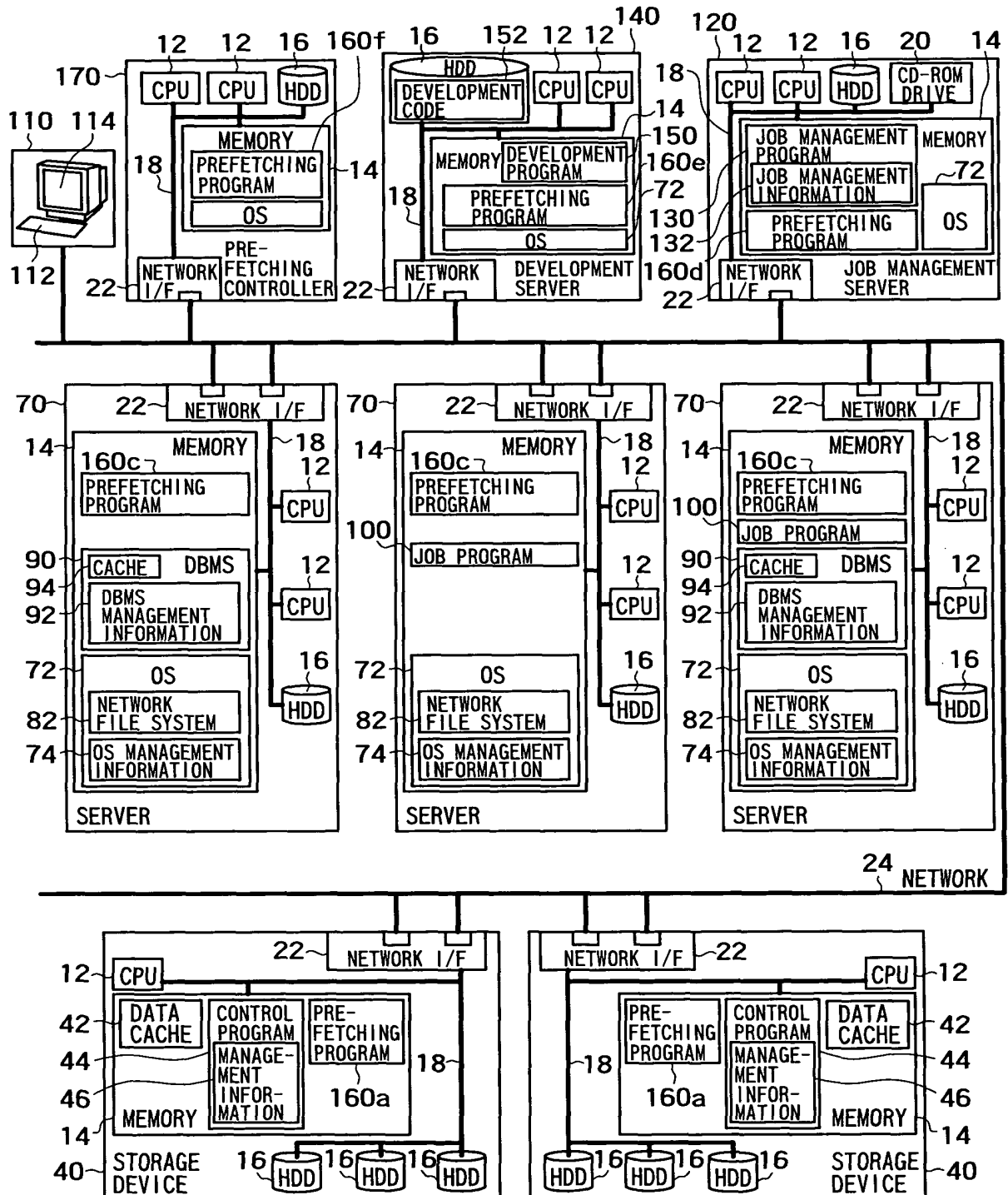


FIG. 21

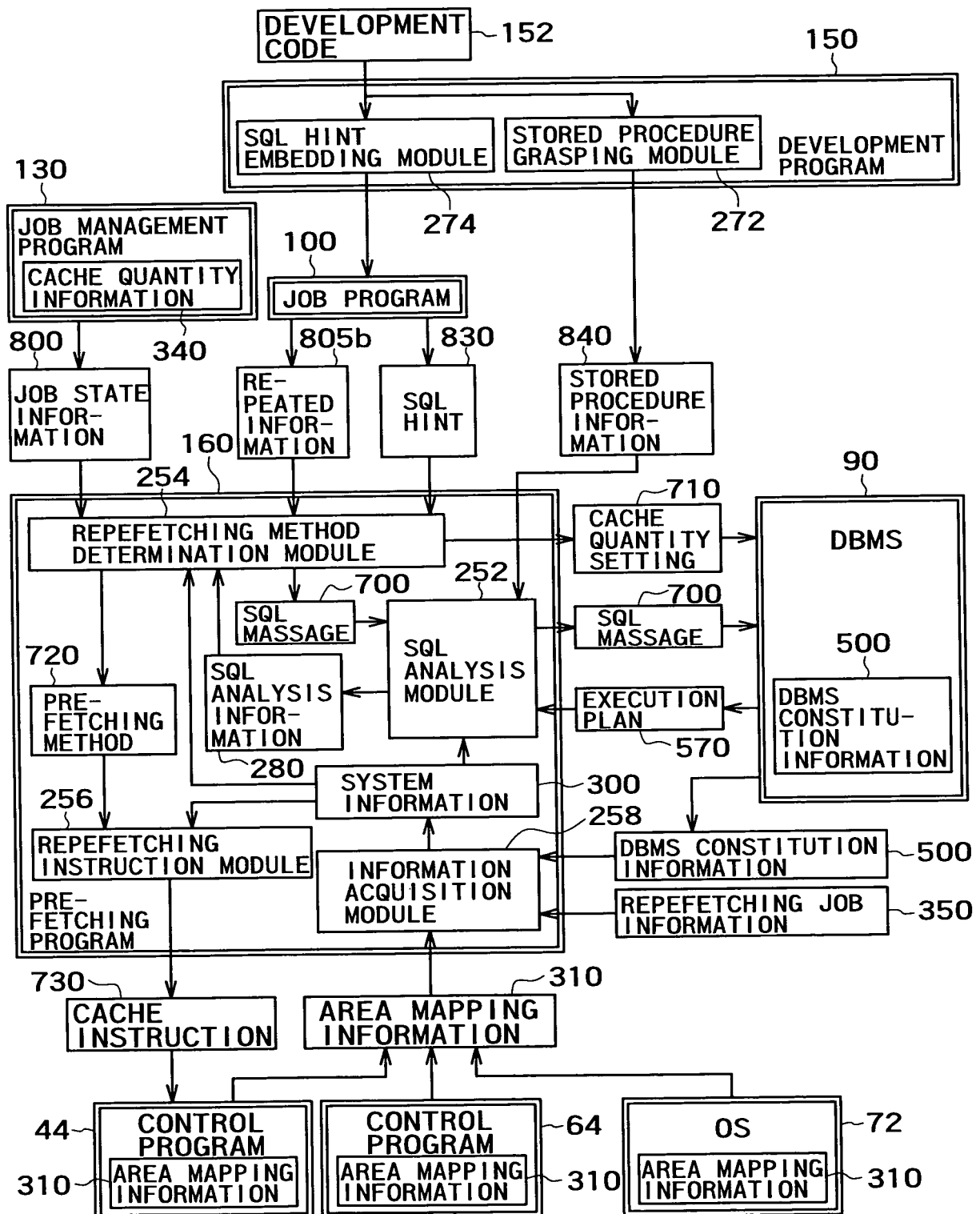


FIG. 22

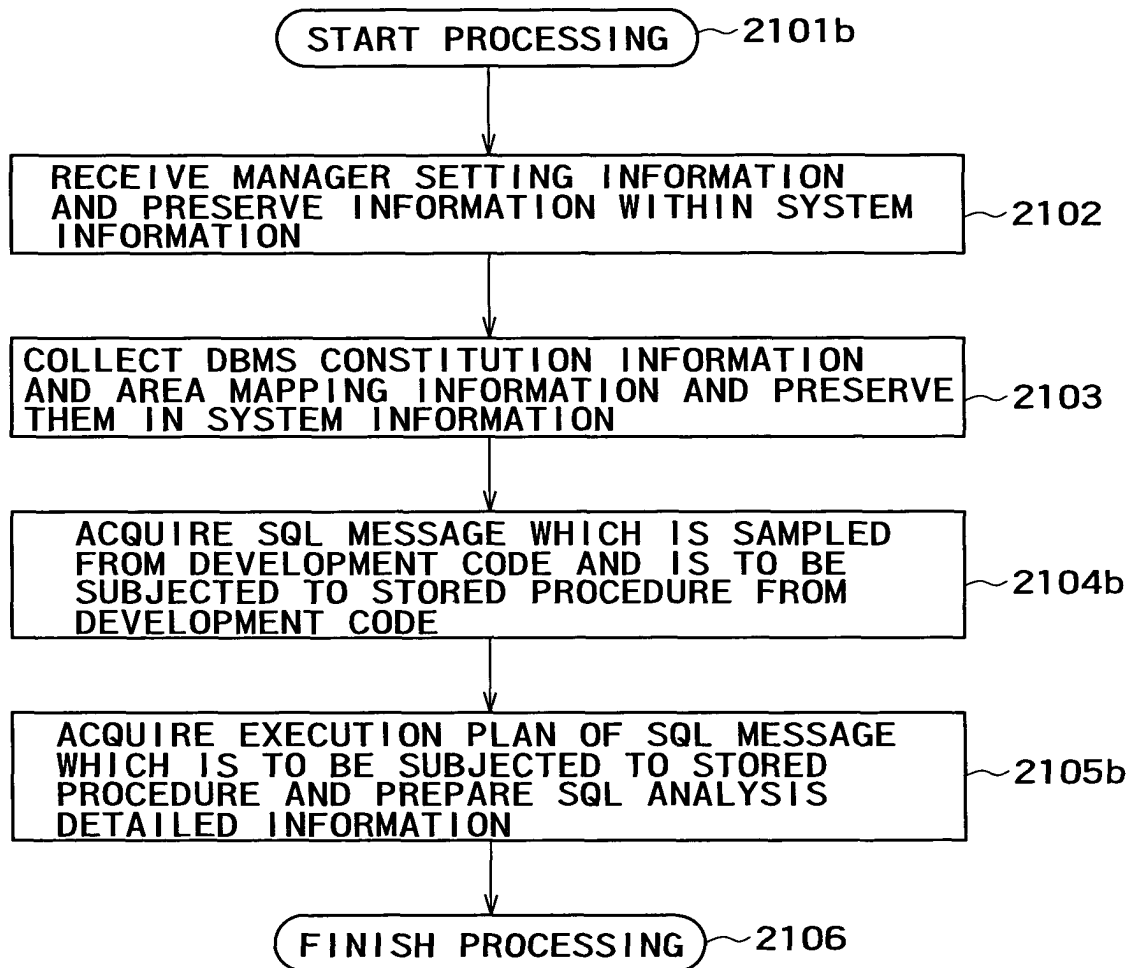


FIG. 23

5202

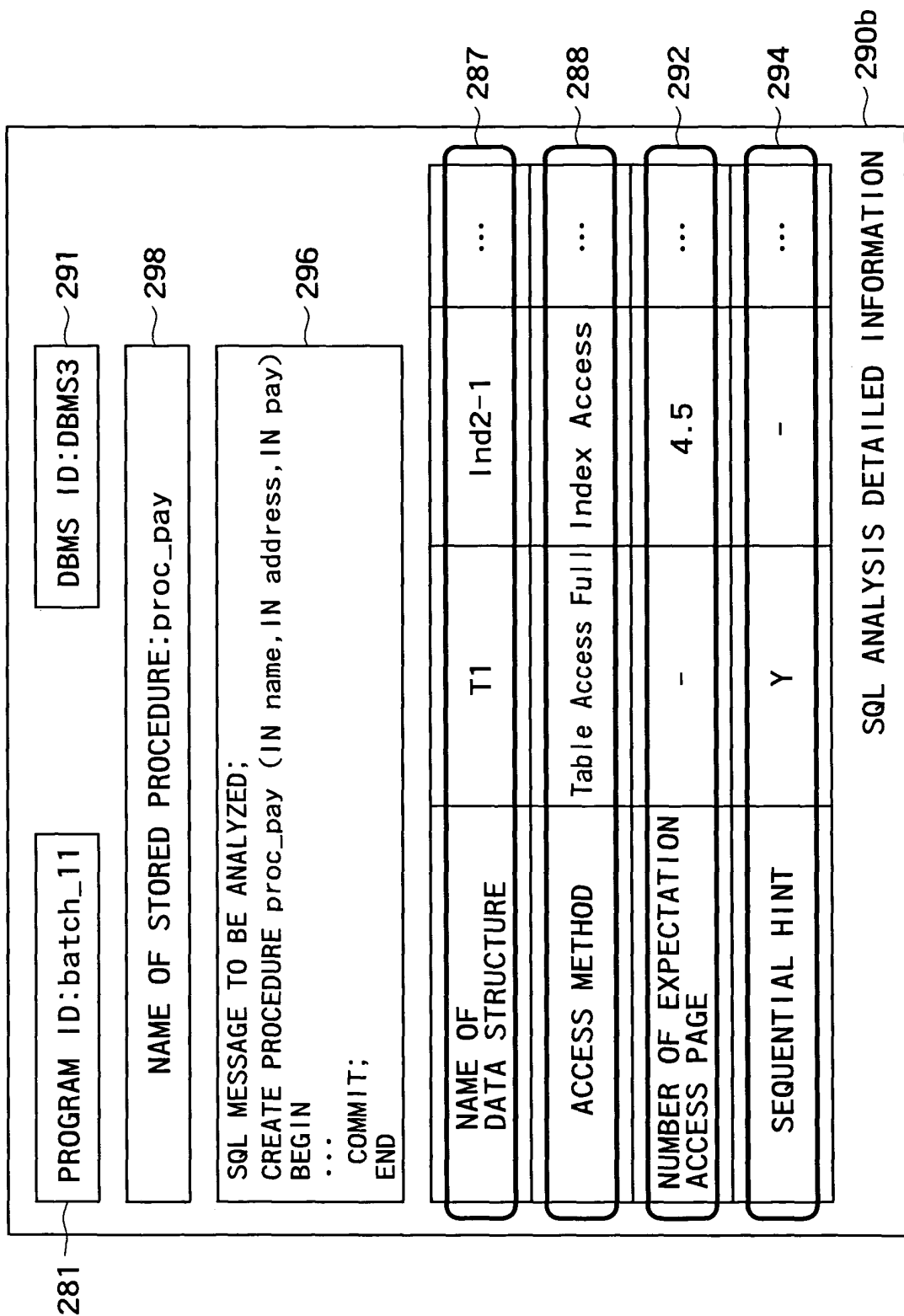
```
CREATE PROCEDURE proc_pay (IN name, IN address, IN pay)
BEGIN
  DECLARE c_id NUMBER(10);
  DECLARE d_id NUMBER(10);

  SELECT C_ID, C_ID INTO :c_id, :d_id
  FROM Cust
  WHERE C_Name=:name and C_Addr=addr
  FOR UPDATE;

  UPDATE Cust
  SET C_Balance=C_Balance-pay
  WHERE C_ID=:c_id;
  ...
  COMMIT;
END
```

5200

FIG. 24



```

...
for(i=0;i<records;i++){
...
name=in_name[i]; addr=in_addr[i]; pay=i[i];
EXEC SQL SELECT C_ID,C_ID INTO:c_id,:c_d_id
FROM Cust
WHERE C_Name=:name and C_Addr=:addr
FOR UPDATE;

EXEC SQL UPDATE Cust
SET C_Balance=C_Balance-:pay
WHERE C_ID=:c_id;
...
COMMIT;
}

```

5002

HINT EMBEDDING
PROCESSING

5024

```

...
EXEC SQL HINT REPEAT START COUNT (:records);

```

5022

```

EXEC SQL HINT EXEC_SQL

```

```

" SELECT C_ID,C_ID
FROM Cust
WHERE C_Name=dummy1 and C_Addr=dummy2
FOR UPDATE;
...
COMMIT;
"

```

5026

5010

```

for(i=0;i<records;i++){
...
name=in_name[i]; addr=in_addr[i]; pay=i[i];
EXEC SQL SELECT C_ID,C_ID INTO:c_id,:c_d_id
FROM Cust
WHERE C_Name=:name and C_Addr=:addr
FOR UPDATE;

EXEC SQL UPDATE Cust
SET C_Balance=C_Balance-:pay
WHERE C_ID=:c_id;
...
COMMIT;
}

```

5002

```

EXEC SQL HINT REPEAT END;

```

5028

```

...

```

FIG. 26

```
DECLATE Tname_cursor CURSOR FOR
  SELECT name
  FROM Objects
  WHERE type=TABLE;
```

5102

```
FOREACH record IN Tname_cursor
BEGIN
```

5104

```
  SELECT name,size
  FROM Tables
  WHERE name=record.name;
```

5106

```
END;
```

HINT EMBEDDING
PROCESSING

```
DECLATE Tname_cursor CURSOR FOR
  SELECT name
  FROM Objects
  WHERE type=TABLE;
```

5102

```
HINT REPEAT START
```

```
  COUNT( SELECT count(name)
          FROM Objects
          WHERE type=TABLE );
```

5024b

5022b

```
EXEC SQL HINT EXEC_SQL
```

```
  " SELECT name,size
    FROM Tables
    WHERE name=record.name; "
```

5110

5026b

```
FOREACH record IN Tname_cursor
BEGIN
```

5104

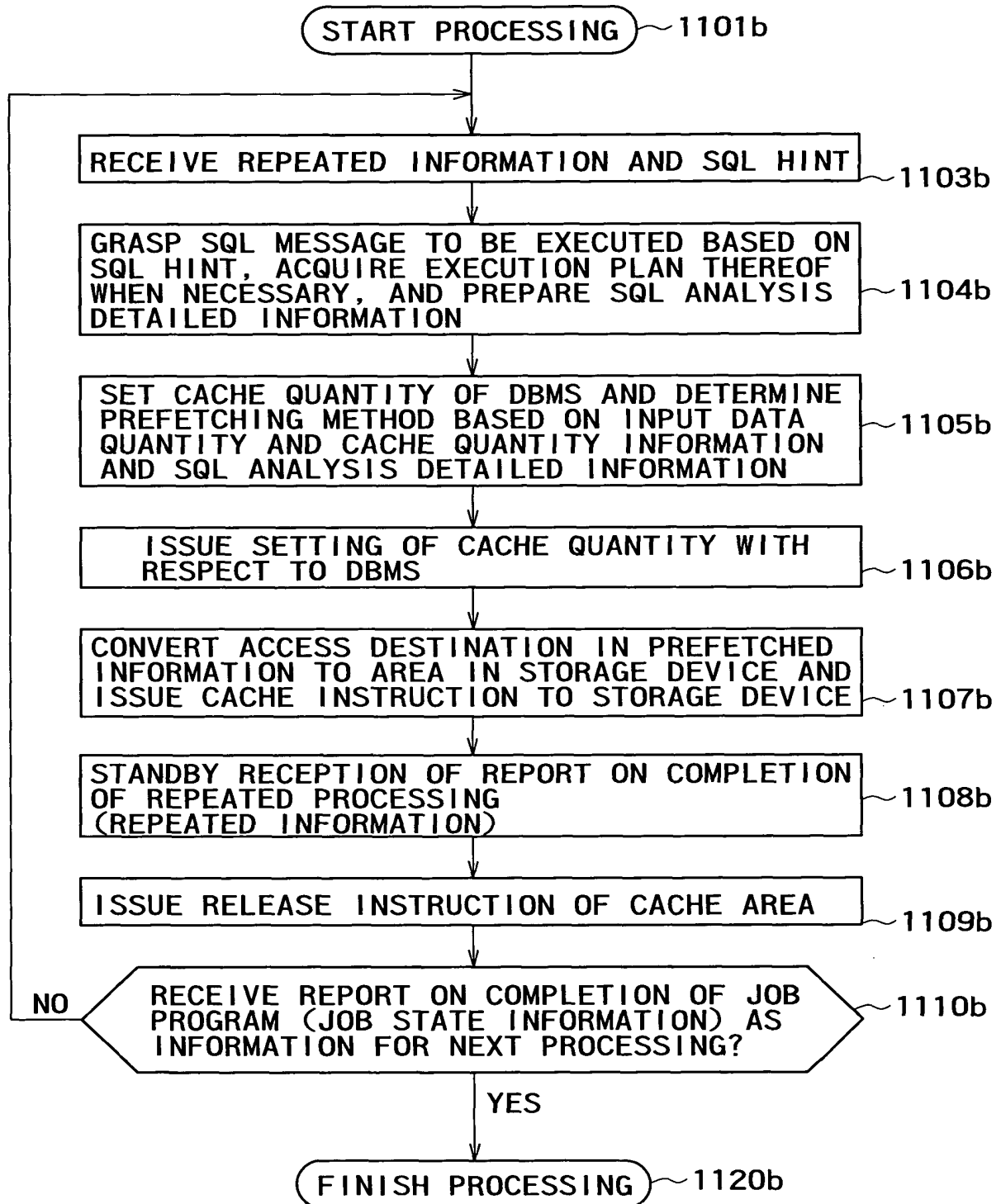
```
  SELECT name,size
  FROM Tables
  WHERE name=record.name;
```

```
END;
```

```
HINT REPEAT END;
```

5028

FIG. 27



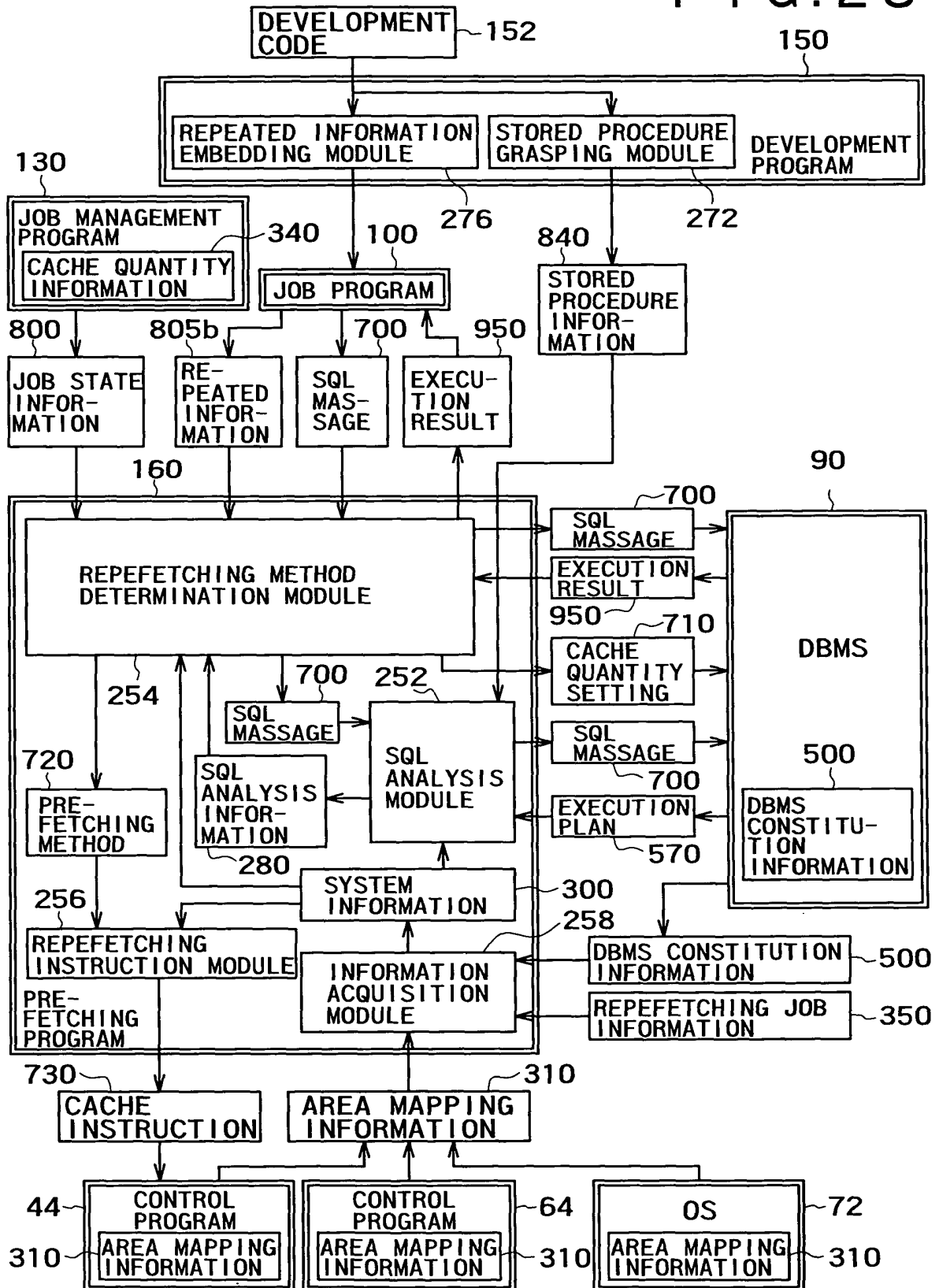


FIG. 29

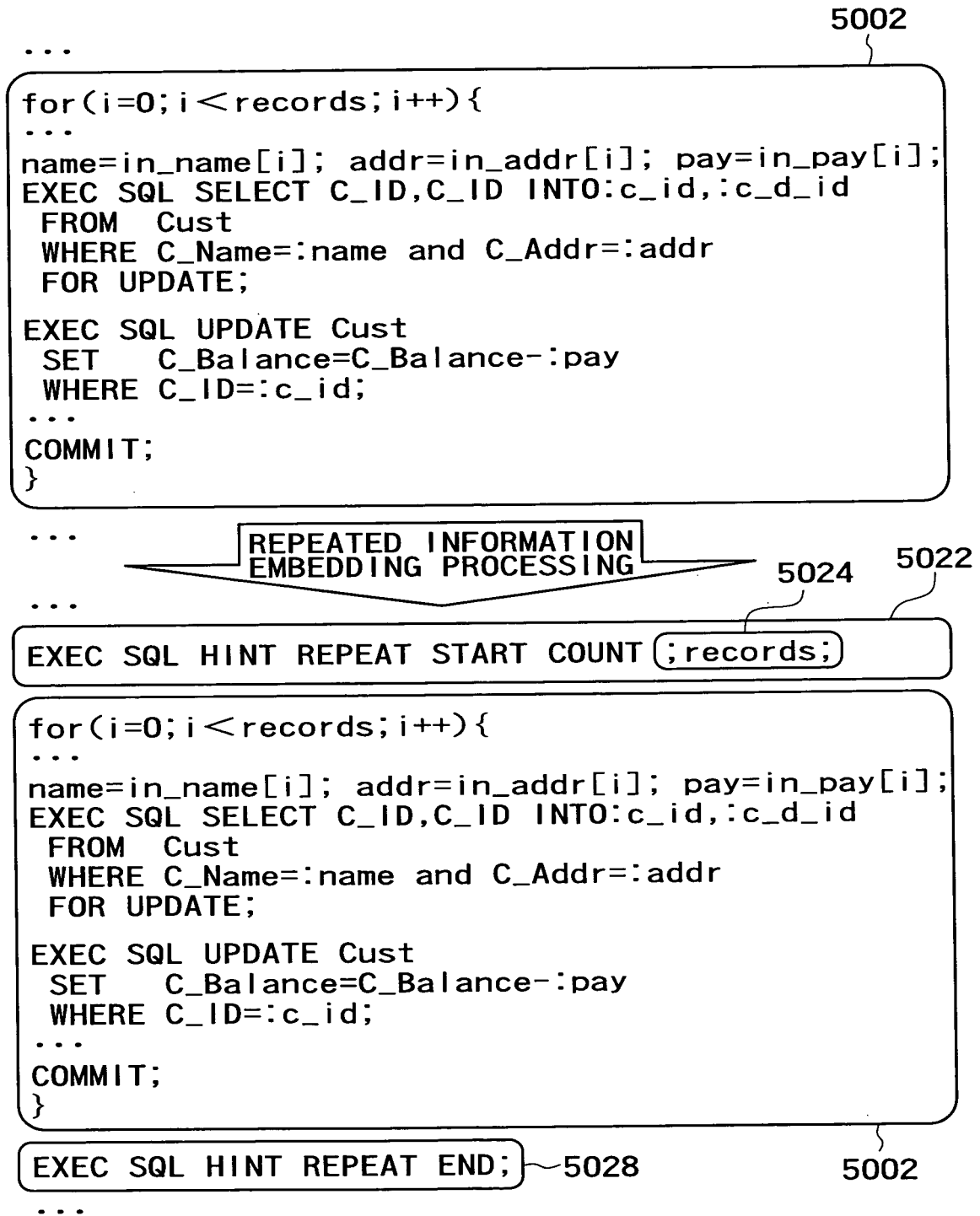


FIG. 30

```
DECLATE Tname_cursor CURSOR FOR
  SELECT name
  FROM Objects
  WHERE type=TABLE;
```

5102

```
FOREACH record IN Tname_cursor
BEGIN
```

5104

```
  SELECT name,size
  FROM Tables
  WHERE name=record.name;
```

5106

```
END;
```

REPEATED INFORMATION
EMBEDDING PROCESSING

```
DECLATE Tname_cursor CURSOR FOR
  SELECT name
  FROM Objects
  WHERE type=TABLE;
```

5102

```
HINT REPEAT START
```

```
  COUNT( SELECT count(name)
          FROM Objects
          WHERE type=TABLE );
```

5024b

5022b

```
FOREACH record IN Tname_cursor
BEGIN
```

5104

```
  SELECT name,size
  FROM Tables
  WHERE name=record.name;
```

```
END;
```

```
HINT REPEAT END;
```

5028

FIG. 31

